REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing this collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. **PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.**

1. REPORT DATE *(DD-MM-YYYY)* 05-18-2015	**2. REPORT TYPE**	**3. DATES COVERED** *(From - To)*

4. TITLE AND SUBTITLE Towards a Personalized Prescription Tool For Diabetic Treatment	**5a. CONTRACT NUMBER**
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S) Dilks, Anna Eileen	**5d. PROJECT NUMBER**
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) U.S. Naval Academy Annapolis, MD 21402	**10. SPONSOR/MONITOR'S ACRONYM(S)**
	11. SPONSOR/MONITOR'S REPORT NUMBER(S) Trident Scholar Report no. 432 (2015)

12. DISTRIBUTION / AVAILABILITY STATEMENT

This document has been approved for public release; its distribution is UNLIMITED.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

For Diabetic patients, insulin is unable to effectively assist in transporting glucose into cells to be used for energy. Type I diabetes arises when the pancreas does not produce enough insulin, and type II diabetes develops when cell receptors become insensitive to insulin. Both conditions present the danger of causing unhealthy glucose levels in the blood stream and are treated with insulin injection therapy to trigger glucose uptake in the cells.

A mathematical model of natural glucose and insulin control allows for a quantitative understanding of the internal glucose-insulin dynamics of healthy and diabetic patients. This research extends the Cobelli model of glucose and insulin dynamics to include both long and short-acting insulin inputs currently used to treat diabetic patients. The project will introduce a personalized approach to treatment by adapting the set of average diabetic Cobelli model parameters over time in response to observed patient feedback data. The personalized model will be combined with a nonlinear model predictive control strategy to determine the best insulin injection routine to achieve healthy glucose levels in diabetic patients. This work will contribute to the development of an individualized prescription tool which physicians can use to more effectively treat diabetic patients.

15. SUBJECT TERMS

insulin, glucose, diabetes, injection, model

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**		40	**19b. TELEPHONE NUMBER** *(include area code)*

A TRIDENT SCHOLAR PROJECT REPORT

NO. 432

Towards a Personalized Prescription Tool For Diabetic Treatment

by

Midshipman 1/C Anna Eileen Dilks, USN

UNITED STATES NAVAL ACADEMY
ANNAPOLIS, MARYLAND

This document has been approved for public release and sale; its distribution is limited.

U.S.N.A. --- Trident Scholar project report; no. 432 (2015)

TOWARDS A PERSONALIZED PRESCRIPTION TOOL FOR DIABETIC TREATMENT

by

Midshipman 1/C Anna Eileen Dilks
United States Naval Academy
Annapolis, Maryland

____
(signature)

Certification of Adviser(s) Approval

Senior Professor Richard O'Brien
Weapons and Systems Engineering Department

____
(signature)

____
(date)

Acceptance for the Trident Scholar Committee

Professor Maria J. Schroeder
Associate Director of Midshipman Research

____
(signature)

____
(date)

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)* 05-18-2015	**2. REPORT TYPE**	**3. DATES COVERED** *(From - To)*

4. TITLE AND SUBTITLE
Towards a Personalized Prescription Tool For Diabetic Treatment

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Dilks, Anna Eileen

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
U.S. Naval Academy
Annapolis, MD 21402

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
Trident Scholar Report no. 432 (2015)

12. DISTRIBUTION / AVAILABILITY STATEMENT

This document has been approved for public release; its distribution is UNLIMITED.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

For Diabetic patients, insulin is unable to effectively assist in transporting glucose into cells to be used for energy. Type I diabetes arises when the pancreas does not produce enough insulin, and type II diabetes develops when cell receptors become insensitive to insulin. Both conditions present the danger of causing unhealthy glucose levels in the blood stream and are treated with insulin injection therapy to trigger glucose uptake in the cells.

A mathematical model of natural glucose and insulin control allows for a quantitative understanding of the internal glucose-insulin dynamics of healthy and diabetic patients. This research extends the Cobelli model of glucose and insulin dynamics to include both long and short-acting insulin inputs currently used to treat diabetic patients. The project will introduce a personalized approach to treatment by adapting the set of average diabetic Cobelli model parameters over time in response to observed patient feedback data. The personalized model will be combined with a nonlinear model predictive control strategy to determine the best insulin injection routine to achieve healthy glucose levels in diabetic patients. This work will contribute to the development of an individualized prescription tool which physicians can use to more effectively treat diabetic patients.

15. SUBJECT TERMS
insulin, glucose, diabetes, injection, model

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**		40	**19b. TELEPHONE NUMBER** *(include area code)*

Abstract:

Insulin is a hormone which attaches to cell receptors, allowing glucose to leave the blood stream and provide energy to cells. Type I diabetes results when the pancreas does not produce enough insulin to assist the uptake of glucose into cells. In the case of type II diabetes, cell receptors have decreased insulin sensitivity and cannot use insulin to transport glucose into cells properly. Both conditions present the danger of causing unhealthy glucose levels in the blood stream and are treated with insulin injection therapy to trigger glucose uptake in the cells.

A mathematical model of natural glucose and insulin control in the body allows for a deeper understanding of the malfunctions that cause diabetes, as well as a more quantified approach to regulating blood glucose levels with insulin injection therapy. Cobelli et al. presented a simulation model which describes the kinetics of glucose digestion and absorption that occur after a meal.[i]

First, this research seeks to extend the Cobelli model of glucose and insulin dynamics to include both long and short-acting insulin inputs currently used to treat diabetic patients. Secondly, the research will introduce a personalized approach to treatment by adapting the combined model parameters over time in response to observed patient feedback data. As a result, the updated Cobelli model parameters can be fitted to the individual patient. The resulting model can be used in future research to predict the outcomes of different insulin dosages and determine the best treatment option to achieve desired glucose levels in diabetic patients. Consequently this research forms the foundation for a personalized prescription tool to improve diabetic patient care.

Keywords: insulin, glucose, diabetes, injection, model

Enclosures: (1) Cobelli Model Flow Chart
(2) Combined Cobelli Model Equation Layout
(3) Cobelli Parameter Spreadsheet
(4) Randomized Patient Data Matlab Code
(5) Parameter Sensitivity Testing Matlab Code
(6) Parameter Adaptation Matlab Code

Acknowledgements:

I would like to thank Professor Richard O’Brien for his support and guidance throughout this research endeavor. He taught me how to cope with the peaks and valleys of research and treated me as a valued colleague on the Diabetes research team. I would also like to recognize Professor Sarah Noble for fostering my interest in medical research, encouraging me to propose a Trident research, and for remaining a part of our research team. Lastly, I would like thank Dr. Ledys DiMarsico for her interest in this research and her enthusiasm in seeking systems engineering research as a method for improving patient treatment.

Table of Contents:

Understanding Insulin and Diabetes

Human cells use insulin to extract glucose from the bloodstream, using the simple sugar for energy. The hormone insulin is analogous to a key that is needed to “unlock” cells by attaching to cell receptors, allowing glucose to enter. Malfunctions in the regulation of insulin and glucose in the body can give rise to pathological conditions including both type I and type II diabetes. Type I diabetes results when the pancreas does not produce enough insulin to assist the uptake of glucose into cells. Type II diabetes is a condition in which cell receptors have decreased insulin sensitivity and cannot use insulin to intake glucose properly. Both conditions result in abnormal glucose levels in the blood stream, potentially resulting in complications including heart disease, stroke, high blood pressure, blindness, kidney failure, and nervous symptom damage.

Both type I and type II diabetes are currently treated with insulin injection therapy to trigger glucose uptake in the cells. The Juvenile Diabetes Research Foundation launched the Artificial Pancreas Project in 2006 to push for the development of an artificial pancreas.[iv] This implant would work in the place of a type 1 diabetic’s failing pancreas, regulating blood glucose levels in the blood by administering the appropriate dose of insulin.

Project Goals

This project will bridge the gap between current artificial pancreas research and clinical application to improve treatment methods for both type I and type II diabetic patients. While the development of an artificial pancreas is a desirable long-term goal, a physiologically accurate glucose control model has immediate potential as a prescription tool for physicians to use. This prescription tool will help physicians to better understand their specific patient’s condition and determine the best course of insulin therapy needed to achieve healthy blood glucose levels.

Figure 1 outlines the research objectives in a flow chart format. First, the research will improve the clinical relevance of an existing model of glucose dynamics which currently only accounts for the influence of insulin that is naturally secreted from the pancreas. The improved model will include additional insulin inputs currently used in injection therapy. The second objective is to adapt the Cobelli model parameters to investigate a personalized approach to treatment, fine tuning the model to best represent the patient's condition. These two research steps will prepare the combined Cobelli and Insulin models for future use in adaptive model predictive control, building the foundation for a personalized prescription tool for physicians to use. The Juvenile Diabetes Research Foundation states the "development of novel learning/patient specific artificial pancreas control algorithms in closed loop control systems" as one of its top priority areas for the fiscal year of 2013.[v]This research will contribute to an individualized prescription tool that physicians can use to better treat diabetic patients, while furthering the development of a control algorithm that could be applied to a functioning artificial pancreas in the future.

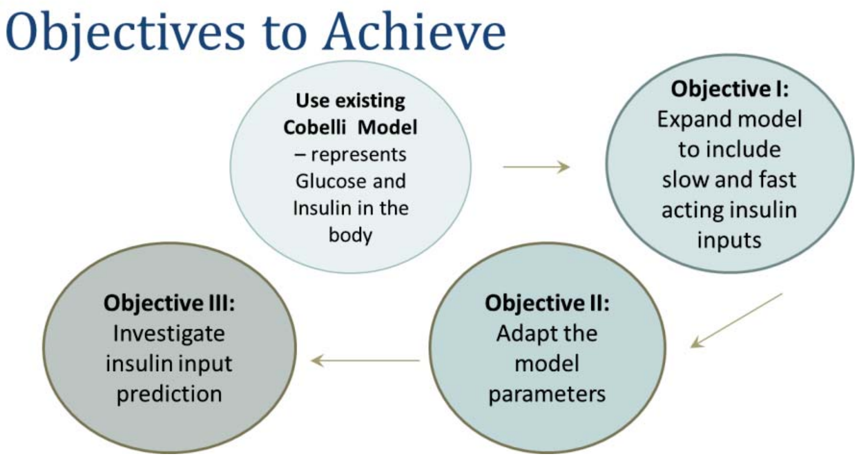

Objectives to Achieve
Use existing Cobelli Model – represents Glucose and Insulin in the body
Objective I: Expand model to include slow and fast acting insulin inputs
Objective II: Adapt the model parameters
Objective III: Investigate insulin input prediction

Figure 1: Research Objectives

Research Timeline

My first semester research goals were centered on developing a working understanding of the Cobelli model, solving the Cobelli differential equations using Simulink, finding an appropriate mathematical model of insulin injections to incorporate into the Cobelli model, and investigating the Cobelli model parameter sensitivity to prepare for parameter adaption.

During second semester, a virtual patient was created using the Cobellii model with a randomized set of model parameters. This generated patient data was used for analysis throughout the research, as patient data from Anne Arundel Medical Center was obtained in early April and could not be used in the project. Parameter sensitivity analysis results identified a subset of the Cobelli model parameters that would be most effective in adapting the published Cobelli diabetic patient parameters to fit the virtual patient data. Additionally, models of slow and fast acting insulin injections were added to the adaptive Cobelli model. This adaptive model will be useful in future research to create a personalized prescription tool to improve the treatment of diabetic patients. The remainder of this report will describe the pertinent background information and results of my work throughout the year.

Cobelli Background and Modeling

Cobelli et al. presented a nonlinear simulation model composed of glucose and insulin subsystems and four unit process models which describe the kinetics and physiological events of glucose digestion and absorption in key organs after a meal. They used a "triple tracer meal protocol" that provided estimates of major glucose and insulin fluxes (i.e. flow rates) during the course of a meal.[vi] The model relates glucose and insulin fluxes to compartmental concentrations in order to characterize the flow through each subsystem, using 28 differential equations to represent glucose and insulin dynamics in a physiologically accurate manner. Cobelli's model is an appealing foundation for my research because a computer simulator of type 1 diabetes was based on this model and approved by the FDA as a substitute to animal trials for preclinical drug testing. [vii]

For the sake of brevity, only the differential equations characterizing the insulin subsystem will be described further. The two-compartment insulin subsystem, displayed in Figure 2, describes the response of the insulin masses in the liver (I_l – *Equation 1*) and plasma ($I_p - Equation\ 2$) to insulin secreted by beta cells in the pancreas (*S*). Specifically, these masses vary as insulin is utilized in the periphery (muscle and tissue) cells to assist in glucose uptake or is lost within the liver. The two equations shown below describe the flow (change in mass over time) of insulin through each compartment. Note that the dot notation indicates the derivative with respect to time. The variables m_1 , m_2 , m_3 , and m_4 are rate parameters describing the exchange between the liver and plasma insulin compartments as well as the rate at which insulin is either lost within the liver or used in surrounding cells.

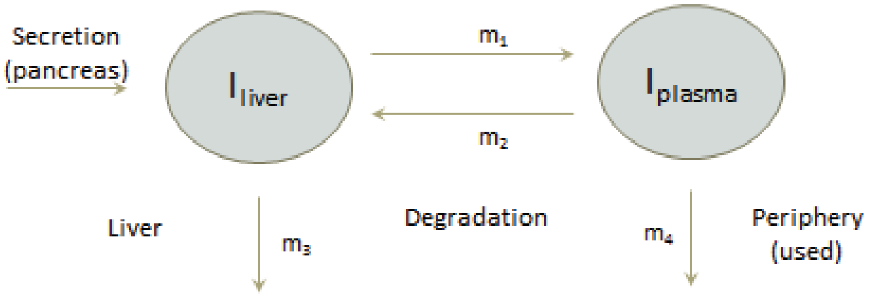


Figure 2: Insulin Subsystem and Equations from Cobelli Model

$$\dot{I}_l(t) = S(t) + \left(m_2 I_p(t) - m_1 I_l(t)\right) - m_3 I_l(t)\ (liver\ insulin)\ \ Equation\ 1$$

$$\dot{I_p}(t) = \left(m_1 I_l(t) - m_2 I_p(t)\right) - m_4 I_p(t)\ (plasma\ insulin)\ \ Equation\ 2$$

Before proceeding, I wanted to possess an in-depth understanding of Cobelli's glucose and insulin model. To do so, I created three reference documents including a block diagram relating the Cobelli model subsystems (Enclosure 1), a similarly structured diagram outlining the necessary equations from Cobelli's original paper (Enclosure 2), and a chart of all model parameters and their physiological meaning (Enclosure 3). MIDN 1/C Brandon Meek, a systems engineering honors student working on a related capstone project, then used these reference documents to build a Simulink model of the Cobelli equations. Simulink is a computer program available at USNA that uses a block diagram (graphical) environment to solve differential equations and simulate dynamic systems over time. Figure 3 shows the Simulink model structure. Each block in this image is a subsystem containing the relevant subsystem equations. The insulin subsystem is circled.

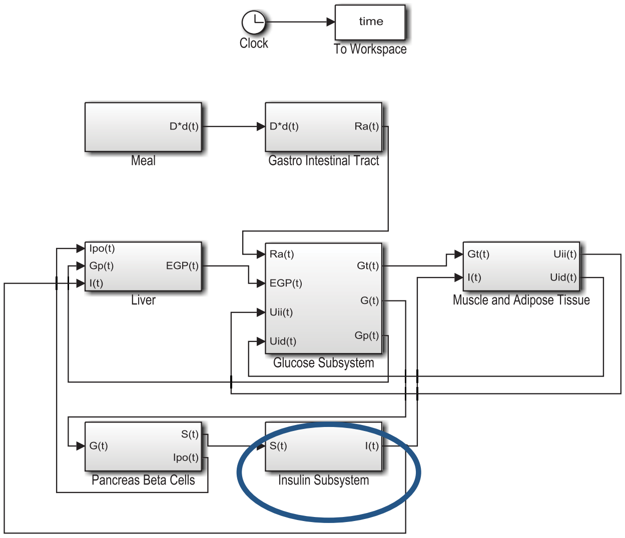


Figure 3: Cobelli Simulink Model

The interior of the insulin subsystem is shown in Figure 4. The two circled integrator, $\left(\frac{1}{s}\right)$, blocks indicate that two differential equations are solved within the subsystem.

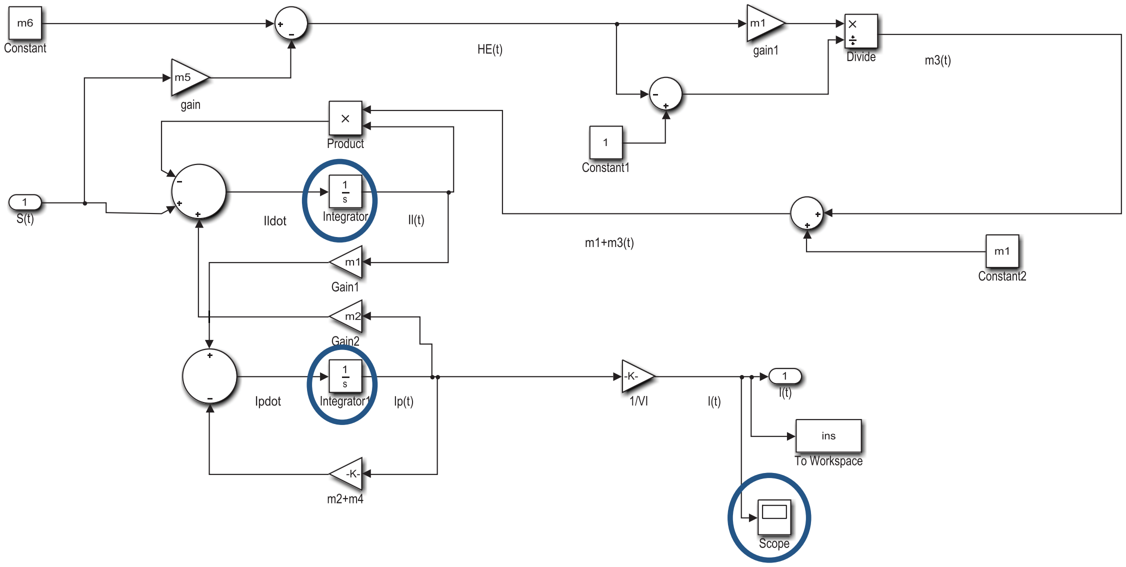


Figure 4: Simulink Insulin Subsystem

The scope allows the user to plot the plasma insulin mass over time and compare the results to those published in Cobelli's paper.[viii] The graph in Figure 5 represents the Simulink insulin data as seen through the MATLAB scope after a patient is given a meal. As expected, the insulin level increases to help transport the ingested glucose from the blood stream into cells to be used for energy. This graph matches the insulin data published by Cobelli, indicating that the Cobelli model can be accurately simulated and manipulated using Simulink code.

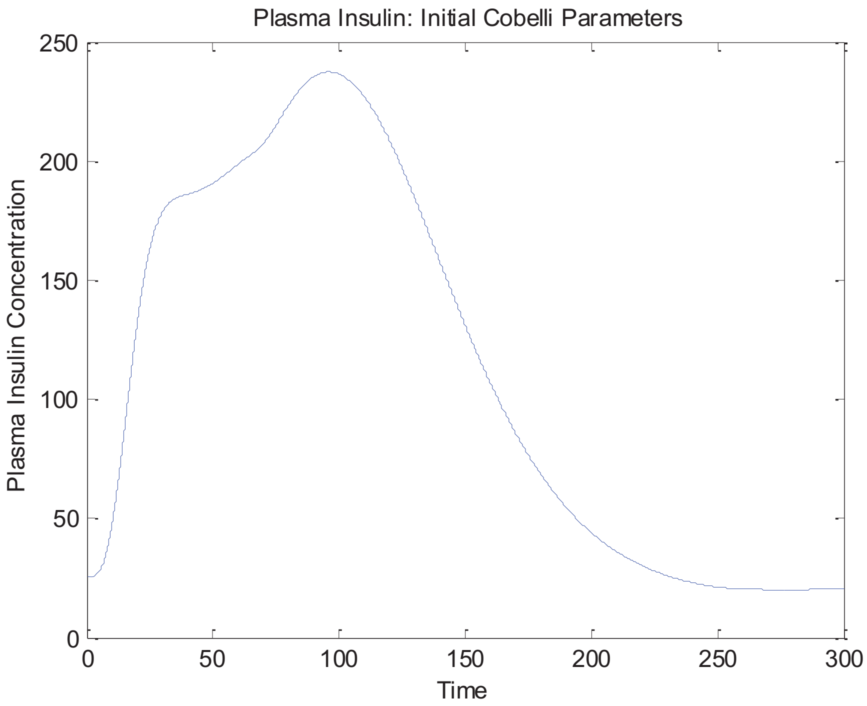


Figure 5: Simulink Insulin Data

Once the Simulink Cobelli model was verified to match the published results, it was used to create a “virtual patient” by randomizing the model parameters. Cobelli et al. published model parameters for both healthy and diabetic patients.[ix] The percent change between these values was calculated, and the parameters were randomized within a range of plus or minus 10% of the diabetic patient parameter value. Using this interval, multiple unique patient parameter sets could be used to generate data using the Cobelli Simulink model. Figure 7 shows the simulation results for the one meal using the original Cobelli parameters, as well as three different patients with randomized parameter values.

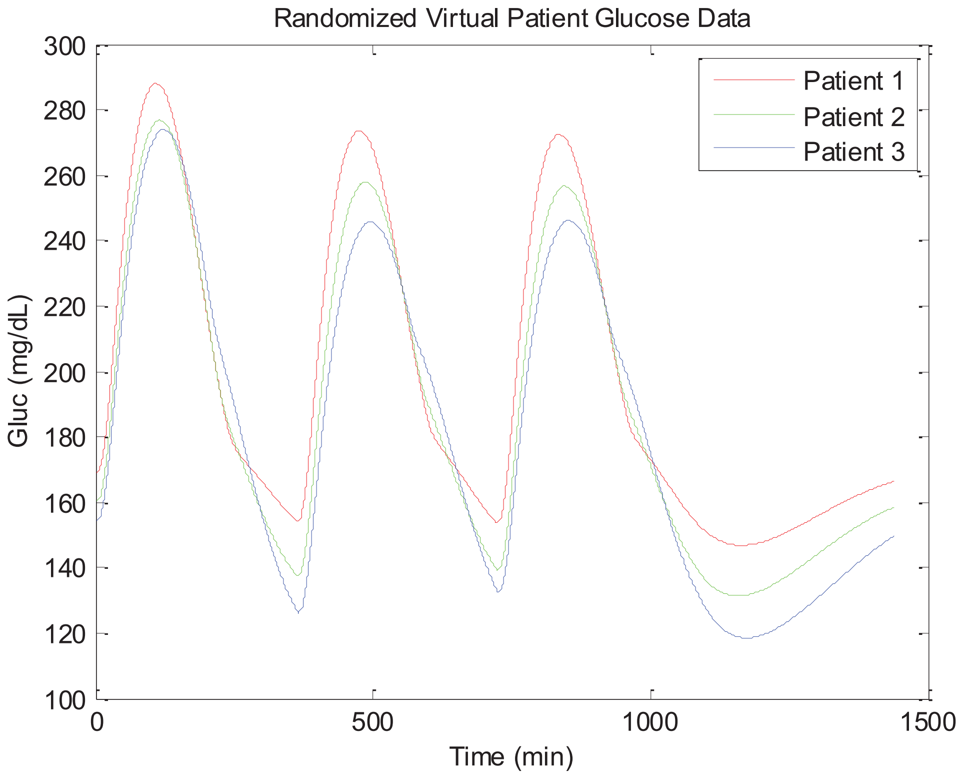


Figure 5: Virtual Patient Data

Objective 1: Improve Clinical Relevancy by Expanding the Cobelli Model

Diabetes patients are currently treated with both rapid-acting and long-acting insulin injections to achieve effective glucose control throughout the course of a day. The characteristics of common insulin analog injections are displayed below.[x]

Rapid Acting	Onset	Peak	Duration
Humalog or Lispro	15-30 min	30-90 min	3-5 hr
Novolog or Aspart	10-20 min	45-50 min	3-5 hr
Apidra or Glulisine	20-30 min	30-90 min	1-2.5 hr

Long Acting	Onset	Peak	Duration
Lantus	1-1.5 hr	None (steady)	20-24 hr
Levemir	1-2 hr	6-8 hr	Up to 24 hr

Table 1: Common Insulin Injection Profiles

The Cobelli model must be modified to include both rapid-acting and long-acting insulin inputs to the plasma insulin compartment. This modification will make the Cobelli model more relevant to doctors considering the model as a potential prescriptive aid. The altered insulin subsystem is shown in Figure 7.

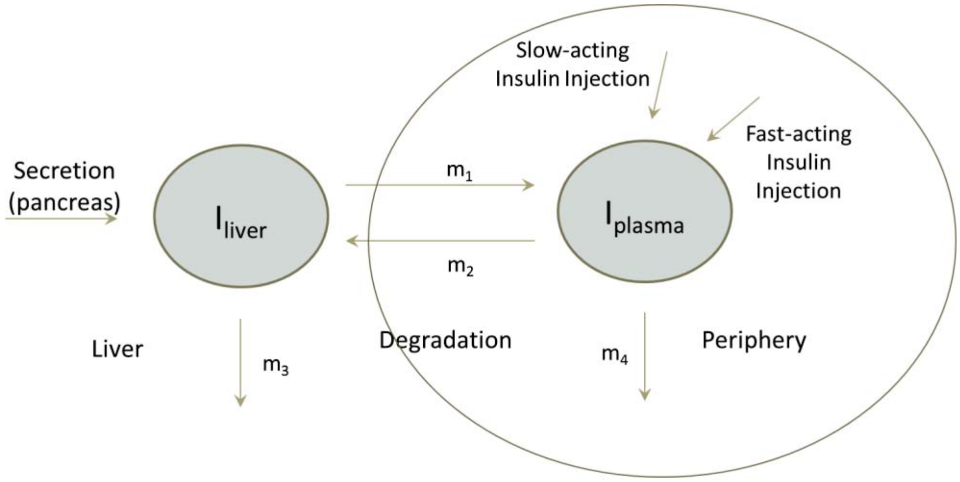


Figure 6: Insulin Subsystem with added injections

The original research proposal suggested the use of first order transfer functionswith differing time constants and an appropriate time delay for rapid- and long-acting insulin injections, as shown in Equation 3 and 4.

$$Slow\ Acting\ Insulin\ Transfer\ Function\quad \frac{I_s(s)}{In(s)} = \frac{1}{\boldsymbol{\tau}_s s + 1} \cdot (e^{-t_d s})\quad Equation\ 3$$

$$Fast\ Acting\ Insulin\ Transfer\ Function\quad \frac{I_f(s)}{In(s)} = \frac{1}{\boldsymbol{\tau}_f s + 1}\quad Equation\ 4$$

However, further literature study revealed significant prior work on the subject of insulin analog injection modeling. The literature trail showed that a single first order equation oversimplifies the injection and absorption of insulin under the skin, through the capillary membrane, and into the bloodstream. A review of four major papers resulted in the final representation chosen. My findings are summarized below.

Similar to our proposal, Huang et al. presents a simple first order model to describe the dynamics of impusive insulin injections.[xi] However, this model does not have separate compartments for the tissue and blood stream with associated time delays. Next, a three compartment model presented by Shimoda et. al. was considered.[xii] This model includes a separate injection depot compartment, which is represented with a first order delay. However, the model only applies to rapid acting insulin, as a first order delay is not enough to simulate the slow dissolving process of long acting injections.

The most helpful paper throughout this literary search was a review of multiple insulin injection models written by G. Nucci and C. Cobelli.[xiii] In this paper, Cobelli dismisses the Shimoda model because it was conceived in the 1980's, before modern insulin analogs were available. A more promising model presented by Trajanoski et. al. describes the breakdown of insulin analogs from an original hexameric state to a dimeric state that can be absorbed into the bloodstream. The model includes an imaginary "bound state" to represent the delayed dynamics of long acting insulin injections. Cobelli praises this model for its chemical accuracy, dependency on both time and injection depth, and inclusion of an inverse relationship between injection volume and absorption rate. At the end of the review paper, Cobelli concludes by suggesting a simplification of the Trajanoski model structure to reduce computational burden.

Jiaxu Li and Yang Kuang answered with their simplified version of Trajanoski's model.[xiv] The model equations vary with time (rather than both time and injection depth), include a biologically accurate Michaelis-Menten description of insulin degradation, and lessen the computational burden of the original model. This model appears to be the best fit to incorporate into Cobelli's model of insulin and glucose dynamics: it describes the chemistry of insulin injection absorption, appropriately accounts for compartmental delays, and has been reviewed positively by Cobelli. The rapid (short) acting and long acting injection model equations shown below were modeled in Simulink to be incorporated into the Cobelli model insulin equation. In these equations, I_{total} represents the total plasma insulin mass, while I_{fast} and I_{slow} represent the individual insulin injection concentrations. [xv]

Short-acting insulin: Lispro

$$H_{fast}(t) = -p\left(H(t) - qD^3(t)\right)\ (hexameric\ form)\quad Equation\ 5$$

$$D_{fast}(t) = p\left(H(t) - qD^3(t)\right) - \frac{bD(t)}{1 + I_{total}(t)}\ (dimeric\ form)\quad Equation\ 6$$

$$I_{fast}(t) = \frac{rbD(t)}{1+I_{total}(t)} - d_i I_{fast}(t)\ (plasma\ concentration\ of\ fast\ insulin)\ Equation\ 7$$

$$H_{fast}(0) = injection\ amount; D_{fast}(0) = 0; I_{fast}(0) = I_{fast}0$$

Long-acting insulin: Glargine

$$B'(t) = -kB(t) * \frac{C_{max}}{1 + H(t)}\ (bound\ form)\quad Equation\ 8$$

$$H_{slow}(t) = -p\left(H(t) - qD^3(t)\right) + kB(t) * \frac{C_{max}}{1 + H(t)}\ (hexameric\ form)\quad Equation\ 9$$

$$D_{slow}(t) = p\left(H(t) - qD^3(t)\right) - \frac{bD(t)}{1 + I_{total}(t)}\ (dimeric\ form)\quad Equaution\ 10$$

$$I_{slow}(t) = \frac{rbD(t)}{1+I_{total}(t)} - d_i I_{fast}(t)\ (plasma\ concentration\ of\ fast\ insulin)\quad Equation\ 11$$

$$B(0) = injection\ amount; D_{slow}(0) = 0; I_{slow}(0) = I_{slow}0$$

Equations 7 and 11 represent the plasma insulin concentration of fast and slow acting insulin which Li and Kuang proposed. When the fast and slow acting insulin models were originally added to the Cobelli plasma insulin equation (represented in Equation 2), the resulting equation was as follows:

$$\dot{I}_{total}(t) = \left(m_1 I_l(t) - m_2 I_p(t)\right) - m_4 I_p(t)\ (plasma\ insulin) + \dot{I}_{fast}(t) + \dot{I}_{slow}(t)\ \ Equation\ 12$$

However, there are two key differences between the Cobelli and Li/Kuang models that are important to note. First, the insulin equations are represented using different units, uU/ml in Li/Kuang and pmol/kg in Cobelli. Additionally, this difference in units reflects the fact that the insulin injection equations represent insulin concentrations in the bloodstream, while the Cobelli equations describe a mass of plasma insulin. To remedy these differences and accurately combine the Cobelli and insulin injection models, the combined insulin equation was altered to multiply each insulin injection term by the total plasma insulin volume in the Cobelli model (Vi). This correction, along with additional unit conversions within the Li/Kuang insulin model parameters, placed the combined equation terms consistently in pmol/kg to represent the combined plasma insulin mass.

A few additional changes were made to the Li/Kuang insulin injection equations to form an appropriate total plasma insulin equation. Originally, the proposed equations for the fast and slow acting insulin injections (Equations 5 and 9) were dependent on the total plasma insulin concentration, both in the insulin addition term and the negative insulin degradation term. The equations were edited so that the addition of dimeric insulin is inversely proportional to the total insulin mass, $I_{total}(t)$, but the degradation of insulin is dependent on the concentration of the injection, $I_{fast}(t)$. The initial condition of the slow and fast acting insulin concentration equations (Equations 7 and 11 –$I_{fast}0$ and $I_{slow}0$) was originally set at 6 uU/ml to normalize the patient's plasma glucose level to basal conditions. This was because patients were given insulin overnight to normalize their plasma glucose levels, isolating the effect of the insulin injections and providing data that would prove useful in modeling the injection dynamics.[xvi][xvii] When the insulin injection concentrations were added to the total insulin mass in Equation 10, the insulin initial conditions were set to zero. The insulin component of the Cobelli model will account for any plasma insulin produced by the body, so no baseline initial condition is needed.

$$\dot{I}_{total}(t) = \left(m_1 I_l(t) - m_2 I_p(t)\right) - m_4 I_p(t)\ (plasma\ insulin) + \dot{I}_{fast}(t) * V_i + \dot{I}_{slow}(t) * V_i$$

$$Equation\ 10$$

The combined model was coded in Simulink and used to administer both slow and fast acting insulin to a virtual patient over a 24-hour period. The patient received 40 units of slow acting insulin at time zero, along with his first meal. He then received 8 units of fast acting insulin along with his second meal 6 hours later. Here, 1 unit of insulin equals 6.94$\times 10^6$ pmol/l. Figure 9 shows the resulting glucose levels of the patient, as compared to the same patient who received no insulin or only the slow insulin injection at time zero. When slow insulin is administered, it lowers the patient's peak and resting glucose levels as shown by the red line graph. The fast insulin, administered along with meal two, creates an additional decrease in the blood glucose level that could not be achieved with slow insulin alone. This is best seen by comparing the red (slow and fast insulin) and green (slow insulin only) lines in in the second spike of glucose concentration. In this case, the patient was given five times the amount of slow acting insulin as compared to fast insulin, so it is not surprising that the fast acting insulin does not have a large impact on the patient's glucose levels.

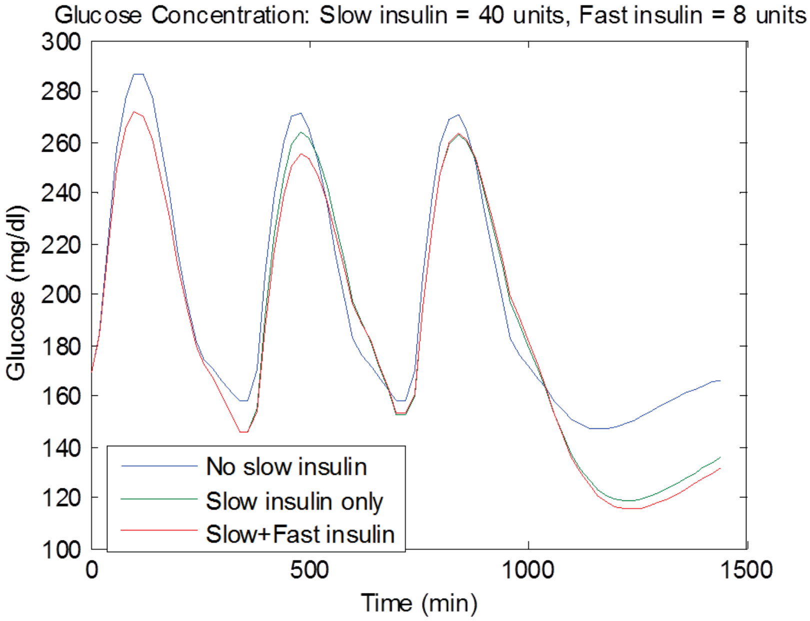


Figure 9: Combining Cobelli and Insulin Models

Figure 10 shows the insulin concentration for the same patient, as slow insulin is administered at time 0 and fast insulin is administered six hours later. The injected insulin is necessary to help the ingested glucose leave the blood stream and enter the cells following a meal. Focusing on the valleys following each meal, the insulin injections prevent the patient's insulin levels from dropping as low as they would without injected insulin. These increased insulin levels assist in glucose uptake, lowering plasma glucose levels between meals.

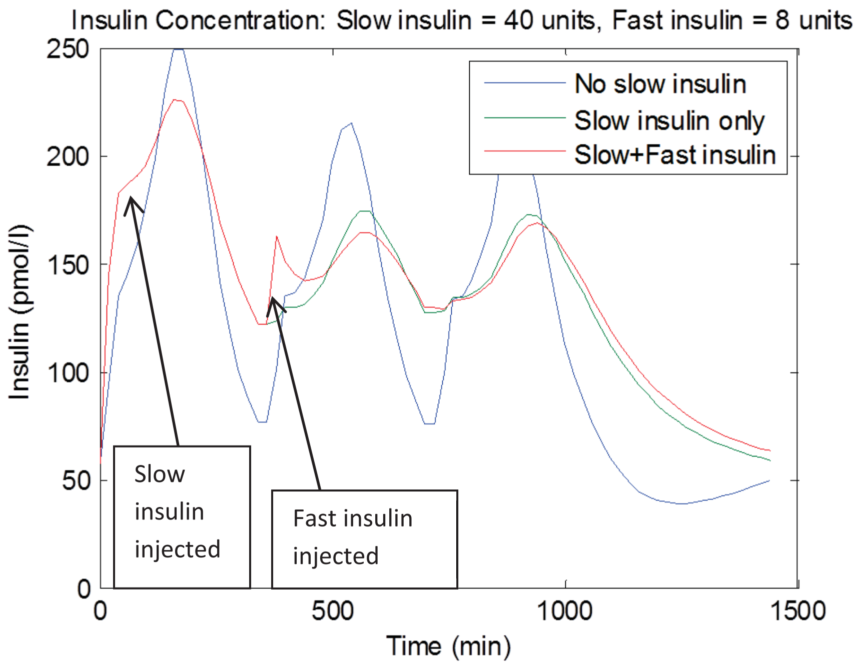


Figure 10: Showing Insulin Concentration changes

This combined model seems to accurately represent the desired effect of injecting a patient with external insulin: lowering blood glucose levels, particularly between meals. These results will need to be compared to patient glucose readings over the course of insulin treatment to verify the accuracy of the combined model.

Objective 2: Adapt the Cobelli Model Parameters

Once the Cobelli model had been verified with Simulink and combined with an appropriate insulin injection model, my focus shifted to preparing to adapt the original Cobelli model parameters to best fit the condition of an individual patient. The first step was to evaluate the sensitivity of the glucose and insulin responses of the Cobelli model to parameter changes. This analysis provided a subset of the most sensitive model parameters that would be most effective in parameter adaptation to create a personalized Cobelli model. The MATLAB code in Enclosure 4 changes each of the 35 model parameters by incremental percentages of their original value, runs the Cobelli Simulink model, and obtains the resulting glucose data. Then, the code calculates the error between the altered and original response to compute the model's sensitivity to the parameter change. The average sensitivity was determined for each parameter using equation 11.

$$Sens = 100 * \frac{\sqrt{\sum \frac{(altered - original)^2}{\# \, readings}}}{\max(gluc) - \min(gluc)} * \frac{1}{parameter\ change} \quad Equation\ 11$$

Several different methods of parameter sensitivity testing were used to ensure the proper parameters were chosen for adaptation. The sensitivity was first calculated using error in resulting glucose data, followed by the insulin data error, as well as a sum squared error combining both glucose and insulin results. Four parameters (Fcns, gamma, Heb, and f) did not change between healthy and diabetic patients in Cobelli's paper, so these parameters will not be adapted to fit patient data and have been removed from all data tables below. Table 2 shows the parameters which were most sensitive when both insulin and glucose error were considered.

Parameter Name	Parameter Description	Average Sensitivity
kp1	EGP at zero glucose and insulin	10711.849
Vg	Distribution volume of glucose	1538.5491
kp4	Amplitude of portal insulin action on liver	888.2471
k1	Glucose rate parameter (b/n plasma and tissue)	677.07393
Vi	Distribution volume of insulin	474.51345
m6	Rate parameter for Hepatic extraction	382.68959
m4	Insulin rate parameter (periphery degredation)	284.53814

Table 2: Parameter Sensitivity Results – Combining Insulin and Glucose Error

Since the parameter sensitivity is determined using glucose and insulin data, it is logical for the most sensitive parameters to be involved with endogenous glucose production from the liver (kp1 and kp4), parameters within the glucose subsystem (Vg, k1), or within the insulin subsystem (Vi, m4, and m6).

After this sensitivity testing was conducted, additional sensitivity testing was designed to not only increase the parameter values and determine resulting sensitivity, but to decrease the parameter values as well. This sensitivity testing yielded the results shown in Table 3. These results have overlapping sensitive parameters with those in Table 2, except for the three parameters highlighted in blue.

Parameter Name	Parameter Description	Average Sensitivity
kp1	EGP at zero glucose and insulin	5777.668044
m5	Rate parameter for Hepatic extraction	1722.260222

kabs	Rate constant of intestinal absorption	977.5885681
kgri	Rate of grinding	795.9737116
Vg	Distribution volume of glucose	533.1015597
kp4	Amplitude of portal insulin action on liver	474.7515014
k1	Glucose rate parameter (b/n plasma and tissue)	416.7587145
m6	Rate parameter for Hepatic extraction	344.0057034

Table 3: Parameter Sensitivity Results - Using + and - parameter changes

After this first two rounds of parameter sensitivity testing, it was decided that parameter adaptation would be evaluated first by adapting the top five parameters from the combined insulin and glucose sensitivity testing (kp1, Vg, kp4, k1, VI). Then, parameter adaptation would be evaluated when ten parameters are adapted, including the top seven combined sensitivity results as well as the three additional parameters highlighted in table 3.

The parameter adaptation code (Enclosure 5) is a Matlab script which uses an optimization function to choose the best set of parameters to match patient glucose data. The script runs through multiple iterations, trying different parameter sets within a determined set of upper and lower bounds, solving the Cobelli model equations, and computing the sum squared error (or cost) between the simulated data and the provided patient data. In this case, the patient data was generated using a Simulink model with randomized parameters, as described before with the best fit. Once the program can no longer lower the cost, it returns the best fit set of parameters. The optimization function also returns the cost between the patient glucose data and the data simulated using the chosen parameters. Here, the goal is to reduce the cost without adapting all 35 Cobelli model parameters, thus reducing computational burden.

Figure 11 shows the results of parameter adaptation in which all 35 Cobelli model parameters were adapted to fit the randomized patient data. In this case, the glucose data simulated with the adapted parameters (black line) and the patient data (blue line) completely overlap. This and the extremely low calculated cost prove that adaptation was successful. However, the parameter adaptation code ran for over 10 minutes before determining the optimal set of parameters. In future research, parameter adaptation will be a portion of a larger input prediction routine, and it is unreasonable to allow this much time for adaptation. These results confirmed the need to choose a smaller set of parameters to more efficiently adapt the Cobelli model to fit patient data.

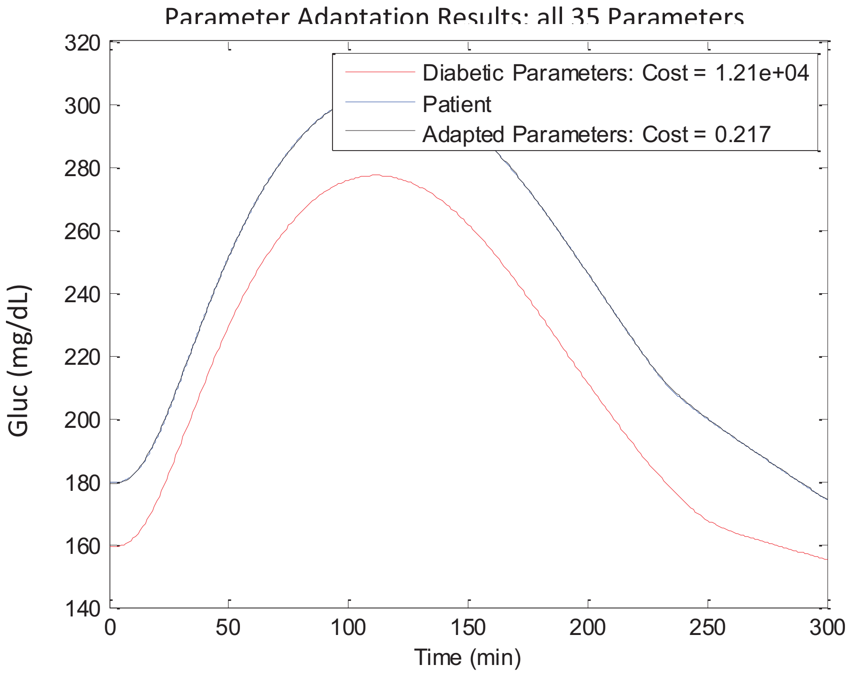


Figure 11: Adapting all 35 parameters

Next, the parameter adaptation code was written to adapt the top five sensitive parameters (as determined from the combined insulin and glucose data) and hold all other parameters constant at the Cobelli values. Doing so greatly increased the cost (from 0.217 to 961). To put the resulting cost in perspective, I compared the parameter adaptation cost (961) to the cost (or sum squared error) between the patient data and the Cobelli model glucose response using the published diabetic patient parameters. The adapted cost is still less than 8% of the cost calculated using the data generated from the original Cobelli parameters; These results are shown in Figure 12 below.

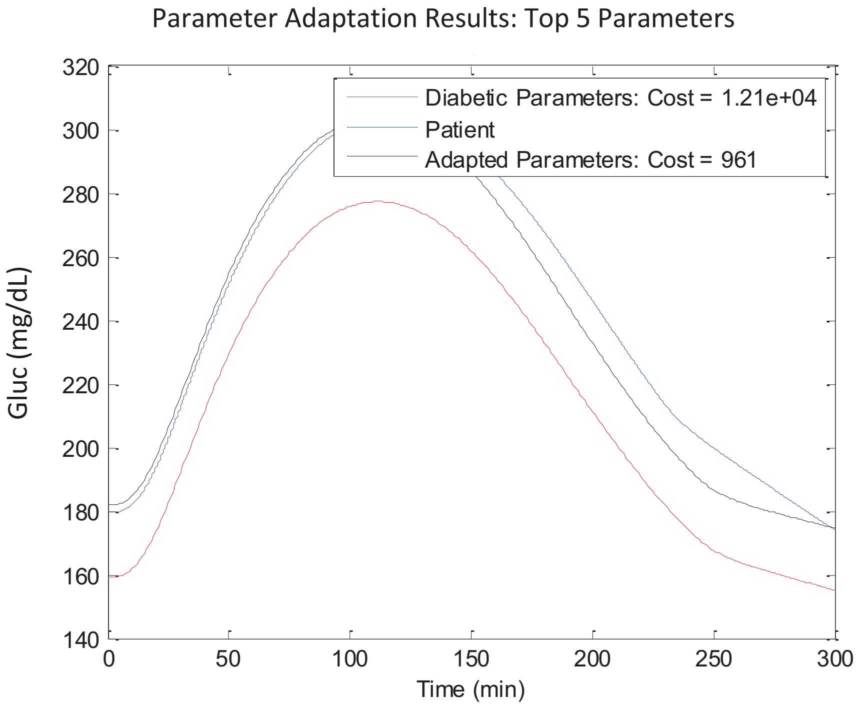


Figure 12: Adapting top 5 parameters (glucose and insulin sensitivity)

The next round of parameter fitting changed ten parameters to fit the randomized patient data. As shown in Figure 13, the cost (161) is significantly reduced by allowing just five more parameters to adapt and fit the patient data. This new cost is 16.7% of the cost resulting from adapting only the top five parameters. The code ran in a reasonable amount of time, suggesting that this is a reasonable number of parameters to adapt in order to personalize the Cobelli model.

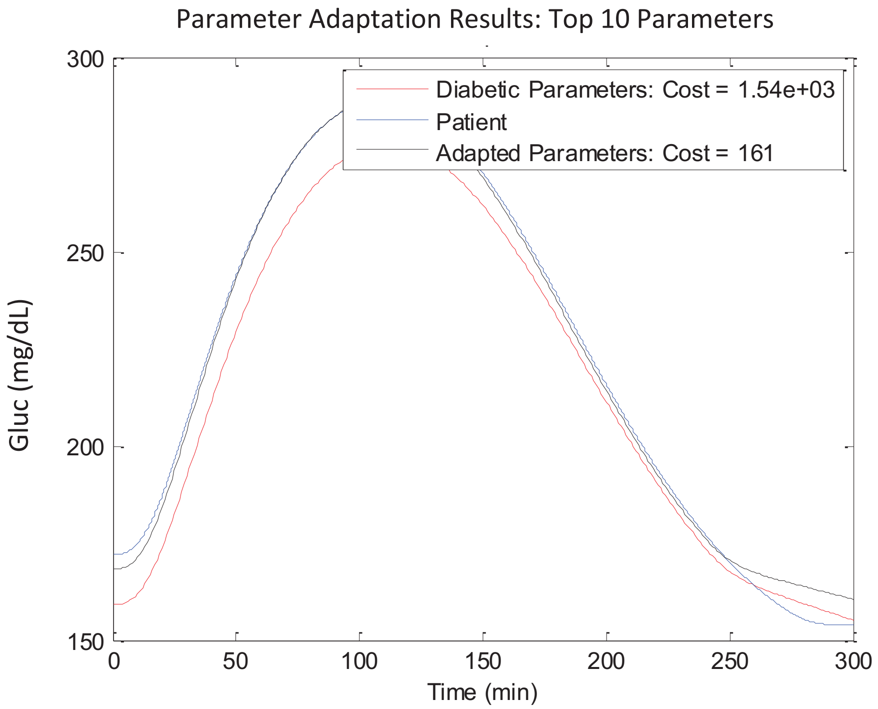


Figure 13: Adapting top 10 Parameters

After these first two rounds of parameter adaption, we revisited the parameter sensitivity testing with a different approach to see if the adaptation results could be improved. This time, instead of altering each parameter by a percentage of its original value, we based the parameter changes on the percent change between the healthy and diabetic value for each parameter. This approach was designed to emulate the method we used to generate the randomized patient data, by altering the original Cobelli diabetic value by a random percentage of the percent change between healthy and diabetic parameter values. Each parameter was changed by positive and negative 5% and 10% of the percent change between healthy and diabetic parameter values. The sensitivity was then calculated using the resulting glucose data, and the top five and ten parameters were chosen for additional rounds of adaptation testing.

Table 4 summarizes the parameter adaptation results for all 5 scenarios: adapting all 35 parameters, the top 5 parameters from the combined glucose and insulin data, top 10 parameters (including positive and negative parameter changes), the top 5 "new" parameters chosen by considering the percent change in parameter values, as well as the top 10 parameters from the same round of parameter sensitivity testing. The first two columns compare the glucose cost calculated using the adapted parameter values to the cost generated using the original diabetic Cobelli parameter values. The new method of parameter sensitivity testing picked out parameters which caused an increased cost as compared to the original method. The original top ten parameters are therefore the most effective in adapting to personalize the Cobelli model to fit patient glucose data.

The third and fourth columns of Table 4 record the sum of the percent error between the adapted and original patient parameters, as well as the percent error between the Cobelli parameters and actual patient parameters. This is a potentially helpful analysis tool because it shows how close the parameters were to matching the randomized patient values. All five cases have very low parameter error percentages, making this a difficult tool to use to distinguish between adaptation methods. Additionally, the parameter percent error did not always decrease when the parameters were allowed to adapt. The most important metric to use when analyzing parameter adaptation effectiveness is the glucose cost, because patient glucose data will be used to personalize the Cobelli model and determine insulin injections in any prescription tool. The individual adapted parameters may not come closer to the patient parameter values, but this is of little importance as long as the simulated glucose data fits the provided patient data with as low of cost as possible.

	Adapted glucose cost	**Diabetic glucose Cost**	**% error adapted parameter**	**%error diabetic parameter**
1 meal, all 35 parameters	0.217	12100	2.07939675	1.983223512
1 meal, 5 parameters	961	12100	1.807557703	1.828632793
1 meal, 10 parameters	119	12100	1.426699217	2.491692
1 meal, 5 new parameters	1630	12100	0.686665424	8.012487486
1 meal, 10 new parameters	1046.7	12100	1.520362334	0.627409478

Table 4: Summarizing Adaptation Results

Conclusion and Future Directions

The two main objectives of this research project were accomplished over the course of the year: the Cobelli model was combined with physiologically accurate models of slow and fast acting insulin injections. The parameters of the Cobelli model equations were adapted to personalize the model, fitting virtual patient glucose readings. With access to patient data from Anne Arundel Medical Center, this combined model can be fit to real patient glucose data to assess parameter adaptation success and verify the model's validity for use as the foundation of a personalized prescription tool. Pending patient data confirmation, this combined adaptive model is prepared for the application of model predictive control to determine appropriate insulin injection treatment methods for diabetic patients.

The topic of Diabetes has become a multigenerational Trident Research focus within the Systems Engineering Department. MIDN Bryan Weisberg from the class of 2013 investigated the addition of glucagon dynamics to the Cobelli model. I built upon his knowledge of the Cobelli model to incorporate insulin input dynamics, create a virtual patient, and apply parameter adaptation to fit randomized patient data. Next year, 1/C Alvin Abes will apply predictive control to this adaptive combined model, working to build a personalized prescription tool which can assist physicians in treating diabetic patients. These continued research efforts under the instruction of Professor Richard O'Brien will further the development of a control algorithm which will improve our understanding of diabetes, improve patient treatment methods with the introduction of mathematical prescription tools, and contribute to the development of an artificial pancreas in the future.

[i] Man, Chiara Dalla, Robert A. Rizza, and Claudio Cobelli. "Meal Simulation Model of the Glucose-Insulin System." *IEEE Transactions on Biomedical Engineering* 54.10 (2007): 1740-749.

[iv] "Artificial Pancreas Project Research." JDRF Improving Lives Curing Type 1 Diabetes. Junior Diabetes Research Foundation, n.d. Web. 04 Feb. 2014.

[v] Ibid.

[vi] Man, Chiara Dalla, Robert A. Rizza, and Claudio Cobelli. "Meal Simulation Model of the Glucose-Insulin System." IEEE Transactions on Biomedical Engineering 54.10 (2007): 1740-749.

[vii] Kovatchev BP, Breton MD, Dalla Man C, Cobelli C. In silico model and computer simulation environment approximating the human glucose/insulin utilization. Food and Drug Administration Master File MAF 1521. 2008

[viii] Man, Chiara Dalla, Robert A. Rizza, and Claudio Cobelli. "Meal Simulation Model of the Glucose-Insulin System." *IEEE Transactions on Biomedical Engineering* 54.10 (2007): 1740-749.

[ix] Ibid.

[x] "Brands and Types of Insulin: Rapid-Acting, Long-Acting, and More." *WebMD*. WebMD, n.d. Web. 01 Dec. 2014.

[xi] Huang, Mingzhan, Jiaxu Li, Xinyu Song, and Hongjian Guo. "MODELING IMPULSIVE INJECTIONS OF INSULIN: TOWARDS ARTIFICIAL PANCREAS." *Society for Industrial and Applied Mathematics* 72.5 (2012): 1524-548. Web.

[xii] Li, Jiaxu, and James Johnson. "MATHEMATICAL MODELS OF SUBCUTANEOUS INJECTION OF INSULIN ANALOGUES: A MINI-REVIEW."*Discrete Continuous Dynamic Systems* (2009): 401-414. Web.

[xiii] Nucci, Gianluca, and Claudio Cobelli. "Models of Subcutaneous Insulin Kinetics. A Critical Review." *Computer Methods and Programs in Biomedicine* (2000): 249-257. Web.

[xiv] Li, Jiaxu, and Yang Kuang. "SYSTEMICALLY MODELING THE DYNAMICS OF PLASMA INSULIN IN SUBCUTANEOUS INJECTION OF INSULIN ANALOGUES FOR TYPE 1 DIABETES." *Mathematical Biosciences and Engineering* (2009): 41-58. Web.

[xv] Ibid.

[xvi] Li, Jiaxu, and Yang Kuang. "SYSTEMICALLY MODELING THE DYNAMICS OF PLASMA INSULIN IN SUBCUTANEOUS INJECTION OF INSULIN ANALOGUES FOR TYPE 1 DIABETES." *Mathematical Biosciences and Engineering* (2009):7. Web.

[xvii] Homko C, Deluzio A, Jimenez C, Kolaczynski JW, Boden G. Comparison of insulin aspart and
lispro. Diabetes Care 2003;26:2027–2031. [PubMed: 12832307]

Bibliography

"Artificial Pancreas Project Research." JDRF Improving Lives Curing Type 1 Diabetes. Junior Diabetes Research Foundation, n.d. Web. 04 Feb. 2014.

"Brands and Types of Insulin: Rapid-Acting, Long-Acting, and More." *WebMD*. WebMD, n.d. Web. 01 Dec. 2014.

Huang, Mingzhan, Jiaxu Li, Xinyu Song, and Hongjian Guo. "MODELING IMPULSIVE INJECTIONS OF INSULIN: TOWARDS ARTIFICIAL PANCREAS." *Society for Industrial and Applied Mathematics* 72.5 (2012): 1524-548. Web.

Homko C, Deluzio A, Jimenez C, Kolaczynski JW, Boden G. Comparison of insulin aspart and lispro. Diabetes Care 2003;26:2027–2031. [PubMed: 12832307]

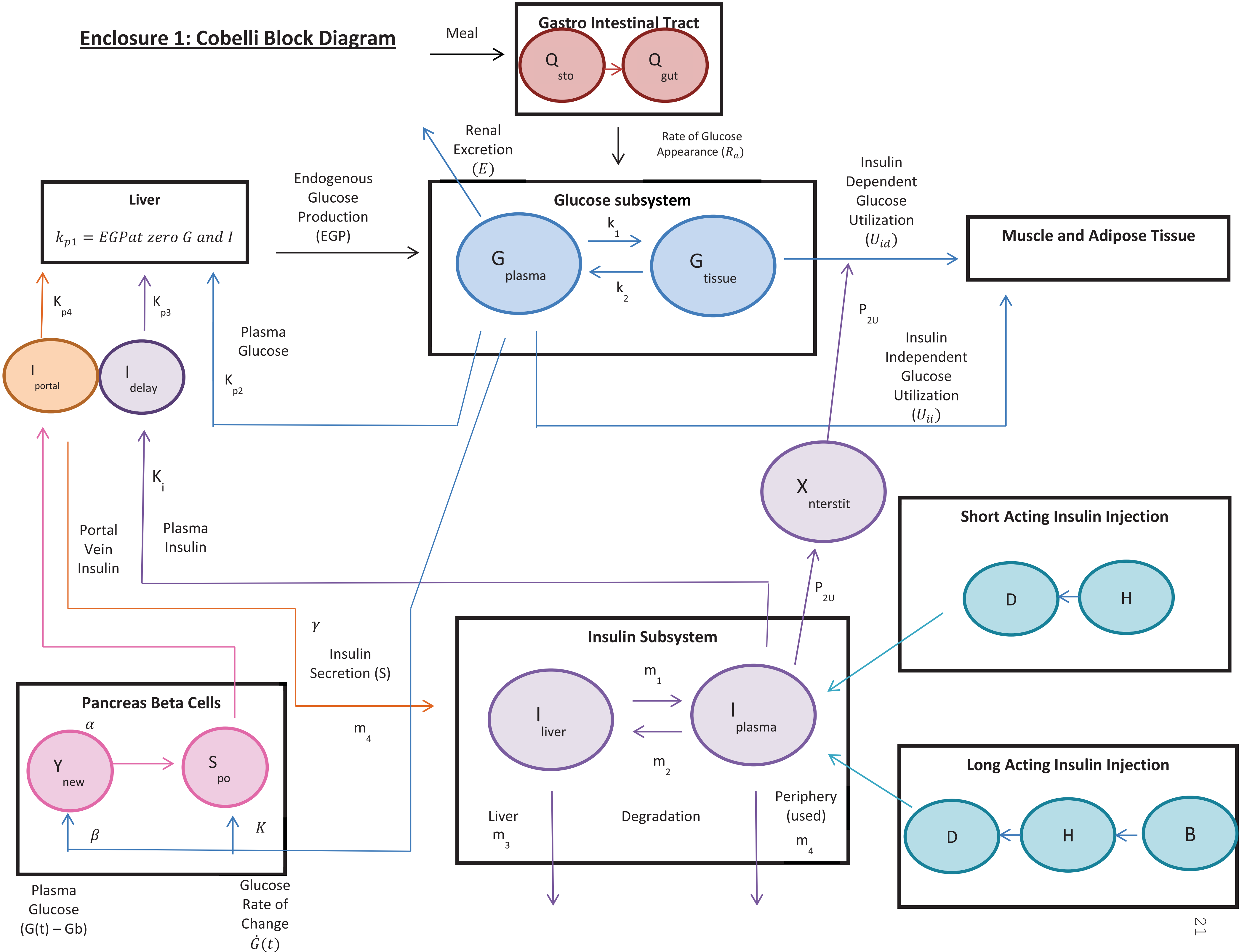
Kovatchev BP, Breton MD, Dalla Man C, Cobelli C. In silico model and computer simulation environment approximating the human glucose/insulin utilization. Food and Drug Administration Master File MAF 1521. 2008

Li, Jiaxu, and James Johnson. "MATHEMATICAL MODELS OF SUBCUTANEOUS INJECTION OF INSULIN ANALOGUES: A MINI-REVIEW."*Discrete Continuous Dynamic Systems* (2009): 401-414. Web.

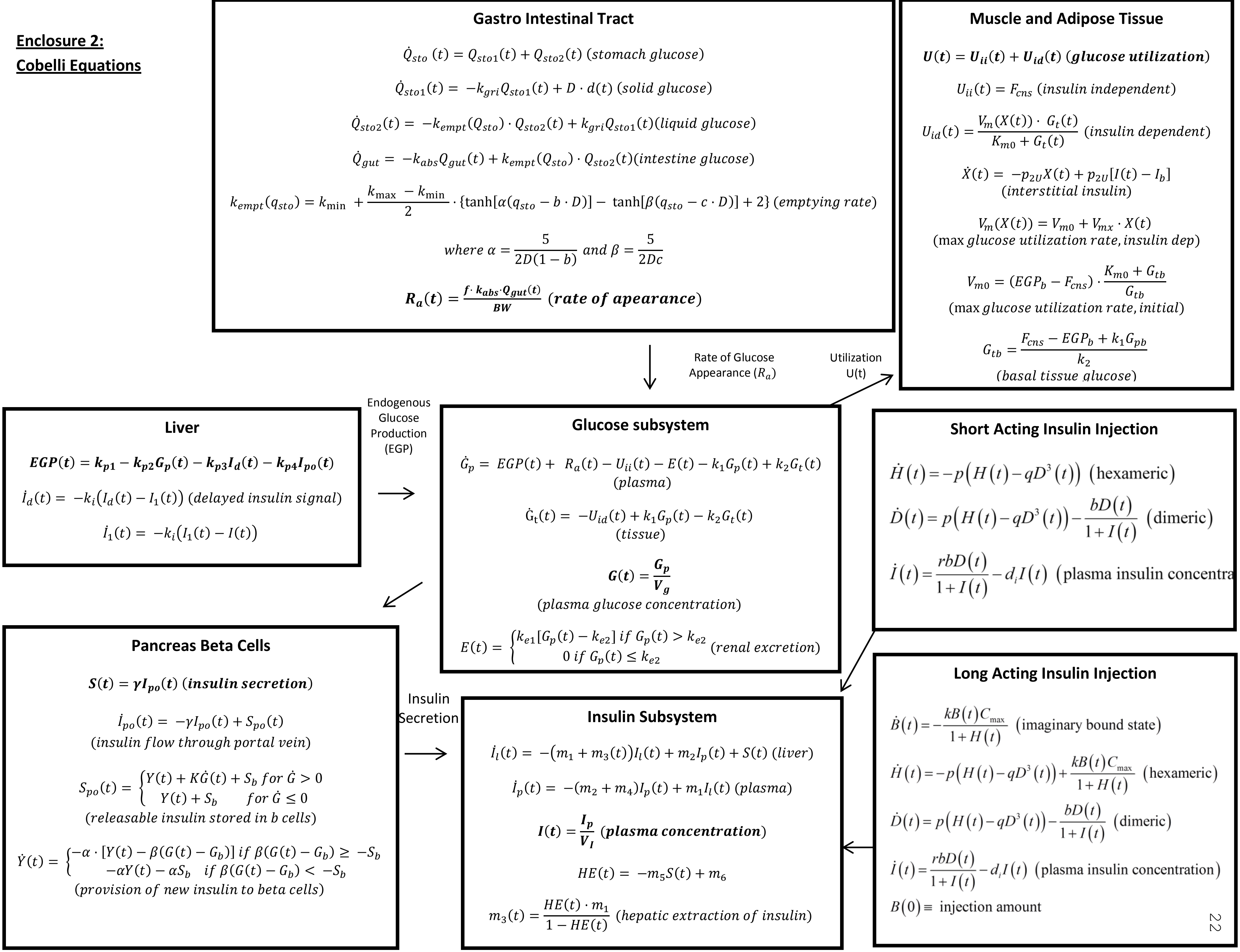
Li, Jiaxu, and Yang Kuang. "SYSTEMICALLY MODELING THE DYNAMICS OF PLASMA INSULIN IN SUBCUTANEOUS INJECTION OF INSULIN ANALOGUES FOR TYPE 1 DIABETES." *Mathematical Biosciences and Engineering* (2009): 41-58. Web.

Man, Chiara Dalla, Robert A. Rizza, and Claudio Cobelli. "Meal Simulation Model of the Glucose-Insulin System." *IEEE Transactions on Biomedical Engineering* 54.10 (2007): 1740-749.

Enclosure 1: Cobelli Block Diagram



Enclosure 2:
Cobelli Equations



Rate of Glucose Appearance (R_a)
Utilization U(t)
Endogenous Glucose Production (EGP)
Insulin Secretion

Gastro Intestinal Tract

$$\dot{Q}_{sto}\,(t) = Q_{sto1}(t) + Q_{sto2}(t)\ (stomach\ glucose)$$

$$\dot{Q}_{sto1}(t) = -k_{gri}Q_{sto1}(t) + D \cdot d(t)\ (solid\ glucose)$$

$$\dot{Q}_{sto2}(t) = -k_{empt}(Q_{sto}) \cdot Q_{sto2}(t) + k_{gri}Q_{sto1}(t)(liquid\ glucose)$$

$$\dot{Q}_{gut} = -k_{abs}Q_{gut}(t) + k_{empt}(Q_{sto}) \cdot Q_{sto2}(t)(intestine\ glucose)$$

$$k_{empt}(q_{sto}) = k_{\min} + \frac{k_{\max} - k_{\min}}{2} \cdot \{\tanh[\alpha(q_{sto} - b \cdot D)] - \tanh[\beta(q_{sto} - c \cdot D)] + 2\}\ (emptying\ rate)$$

$$where\ \alpha = \frac{5}{2D(1-b)}\ and\ \beta = \frac{5}{2Dc}$$

$$\boldsymbol{R_a(t) = \frac{f \cdot k_{abs} \cdot Q_{gut}(t)}{BW}\ (rate\ of\ apearance)}$$

Muscle and Adipose Tissue

$$\boldsymbol{U(t) = U_{ii}(t) + U_{id}(t)\ (glucose\ utilization)}$$

$$U_{ii}(t) = F_{cns}\ (insulin\ independent)$$

$$U_{id}(t) = \frac{V_m(X(t)) \cdot G_t(t)}{K_{m0} + G_t(t)}\ (insulin\ dependent)$$

$$\dot{X}(t) = -p_{2U}X(t) + p_{2U}[I(t) - I_b]$$
$$(interstitial\ insulin)$$

$$V_m(X(t)) = V_{m0} + V_{mx} \cdot X(t)$$
$$(\max glucose\ utilization\ rate, insulin\ dep)$$

$$V_{m0} = (EGP_b - F_{cns}) \cdot \frac{K_{m0} + G_{tb}}{G_{tb}}$$
$$(\max glucose\ utilization\ rate, initial)$$

$$G_{tb} = \frac{F_{cns} - EGP_b + k_1 G_{pb}}{k_2}$$
$$(basal\ tissue\ glucose)$$

Liver

$$\boldsymbol{EGP(t) = k_{p1} - k_{p2}G_p(t) - k_{p3}I_d(t) - k_{p4}I_{po}(t)}$$

$$\dot{I}_d(t) = -k_i\big(I_d(t) - I_1(t)\big)\ (delayed\ insulin\ signal)$$

$$\dot{I}_1(t) = -k_i\big(I_1(t) - I(t)\big)$$

Glucose subsystem

$$\dot{G}_p = EGP(t) + R_a(t) - U_{ii}(t) - E(t) - k_1G_p(t) + k_2G_t(t)$$
$$(plasma)$$

$$\dot{\mathrm{G}}_{\mathrm{t}}(t) = -U_{id}(t) + k_1G_p(t) - k_2G_t(t)$$
$$(tissue)$$

$$\boldsymbol{G(t) = \frac{G_p}{V_g}}$$
$$(plasma\ glucose\ concentration)$$

$$E(t) = \begin{cases} k_{e1}[G_p(t) - k_{e2}]\ if\ G_p(t) > k_{e2} \\ 0\ if\ G_p(t) \le k_{e2} \end{cases}\ (renal\ excretion)$$

Short Acting Insulin Injection

$$\dot{H}(t) = -p\big(H(t) - qD^3(t)\big)\ \ (\text{hexameric})$$

$$\dot{D}(t) = p\big(H(t) - qD^3(t)\big) - \frac{bD(t)}{1 + I(t)}\ \ (\text{dimeric})$$

$$\dot{I}(t) = \frac{rbD(t)}{1 + I(t)} - d_i I(t)\ \ (\text{plasma insulin concentra}$$

Pancreas Beta Cells

$$\boldsymbol{S(t) = \gamma I_{po}(t)\ (insulin\ secretion)}$$

$$\dot{I}_{po}(t) = -\gamma I_{po}(t) + S_{po}(t)$$
$$(insulin\ flow\ through\ portal\ vein)$$

$$S_{po}(t) = \begin{cases} Y(t) + K\dot{G}(t) + S_b\ for\ \dot{G} > 0 \\ Y(t) + S_b\ \ \ \ \ \ for\ \dot{G} \le 0 \end{cases}$$
$$(releasable\ insulin\ stored\ in\ b\ cells)$$

$$\dot{Y}(t) = \begin{cases} -\alpha \cdot [Y(t) - \beta(G(t) - G_b)]\ if\ \beta(G(t) - G_b) \ge -S_b \\ -\alpha Y(t) - \alpha S_b\ \ \ if\ \beta(G(t) - G_b) < -S_b \end{cases}$$
$$(provision\ of\ new\ insulin\ to\ beta\ cells)$$

Insulin Subsystem

$$\dot{I}_l(t) = -\big(m_1 + m_3(t)\big)I_l(t) + m_2I_p(t) + S(t)\ (liver)$$

$$\dot{I}_p(t) = -(m_2 + m_4)I_p(t) + m_1I_l(t)\ (plasma)$$

$$\boldsymbol{I(t) = \frac{I_p}{V_I}\ (plasma\ concentration)}$$

$$HE(t) = -m_5S(t) + m_6$$

$$m_3(t) = \frac{HE(t) \cdot m_1}{1 - HE(t)}\ (hepatic\ extraction\ of\ insulin)$$

Long Acting Insulin Injection

$$\dot{B}(t) = -\frac{kB(t)C_{\max}}{1 + H(t)}\ \ (\text{imaginary bound state})$$

$$\dot{H}(t) = -p\big(H(t) - qD^3(t)\big) + \frac{kB(t)C_{\max}}{1 + H(t)}\ \ (\text{hexameric})$$

$$\dot{D}(t) = p\big(H(t) - qD^3(t)\big) - \frac{bD(t)}{1 + I(t)}\ \ (\text{dimeric})$$

$$\dot{I}(t) = \frac{rbD(t)}{1 + I(t)} - d_i I(t)\ \ (\text{plasma insulin concentration})$$

$$B(0) \equiv \text{ injection amount}$$

Enclosure 3: Cobelli Parameters

	Subsystem	Parameter	Physiological Meaning	Normal Value	Type 2 Diabetic Value	Units
1	Glucose Kinetics	Vg	distribution volume of glucose	1.88	1.49	kl/kg
2		k1	glucose rate parameter (b/n plasma and tissue)	0.065	0.042	1/min
3		k2	glucose rate parameter (b/n plasma and tissue)	0.079	0.071	1/min
4	Renal Excretion	ke1	glomerular filtration rate	0.0005	0.0007	1/min
5		ke2	renal threshold of glucose	339	269	mg/kg
6	Insulin Kinetics	VI	distribution volume of insulin	0.05	0.04	l/kg
7		m1	insulin rate parameter (b/n liver and plasma)	0.19	0.379	1/min
8		m2	insulin rate parameter (b/n liver and plasma)	0.484	0.673	1/min
		m3	time dependent rate parameter (liver degredation)			
9		m4	insulin rate parameter (periphery degredation)	0.194	0.269	1/min
10		m5	Related to Hepatic Extraction	0.0304	0.0526	min*kg/pmol
11		m6	Related to Hepatic Extraction	0.6471	0.8118	dimensionless
12		Heb	Hepatic extraction of insulin	0.6	0.6	dimensionless
13	Rate of Glucose Appearance	kmax	glucose)	0.0558	0.0465	1/min
14		kmin	minimum kemp	0.008	0.0076	1/min
15		kabs	rate constant of intenstinal absorption	0.057	0.023	1/min
16		kgri	rate of grinding	0.0558	0.0465	1/min
17		f	fraction of intestinal absorption which appears in plasma	0.9	0.9	dimensionless
18		a		0.00013	0.00006	
19		b	percentage of dose for which kemp decreases (flex point	0.82	0.68	
20		c	percentage of dose for which kemp recovers (flex point of	0.00236	0.00023	
21		d		0.01	0.09	
22	Endogenous Glucose Production	kp1	EGP at zero gluclose and insulin	2.7	3.09	mg/kg/min
23		kp2	liver glucose effectiveness	0.0021	0.0007	1/min
24		kp3	amplitude of insulin action on liver	0.009	0.005	pmol/l
25		kp4	amplitude of portal insulin action on liver	0.0618	0.0786	pmol/kg
26		ki	delay between insulin signal and action	0.0079	0.0066	1/min

Enclosure 3: Cobelli Parameters

27	**Glucose Utilization**	Fcns	glucose uptake by brain and erythrocytes	1	1	mg/kg/min
28		Vm0	initial max utilization rate	2.5	4.65	mg/kg/min
29		Vmx	max utilization rate, insulin dependent	0.047	0.034	pmol/l
30		Km0	(not dependent on insulin)	225.59	466.21	mg/kg
31		P2U	utilization	0.0331	0.084	1/min
32	**Insulin Secretion**	K	pancreatic responsivity to the glucose rate of change	2.3	0.99	pmol/kg per
33		α	delay between glucose signal and insulin secretion	0.05	0.013	1/min
34		β	pancreative responsitivty to glucose	0.11	0.05	pmol/kg/mirn per
35		γ	transfer rate bewteen portal vein and liver	0.5	0.5	1/min
36	**Li Short Acting (Lispro)**	p	trasform rate from 1 heameric to 3 dimeric molecules		0.5	1/min
37		q	chemichal equilbration constant		0.13	ml^2/U^2
38		b	absorption constant parameter (Lispro)		0.0068	U/min
39		r	shell radius?		0.2143	cm
40		di	insulin degredation rate (Lipsro)		0.081	1/min
41		H(0)	Injection amount		2.90E+03	uU/mL
42	**Li Long Acting**	k	rate of hexamer disengagement from bound state		2.30E-05	1/min
43		Cmax	maximum capacity of transformation		15	U/mL
44		p	trasform rate from 1 heameric to 3 dimeric molecules		0.5	1/min
45		q	chemichal equilbration constant		3.04	ml^2/U^2
46		b	absorption constant parameter		0.02	U/min
47		r	shell radius?		0.2143	cm
48		di	insulin degredation rate		0.0215	1/min
49		B(0)	injection amount		6.70E+03	nU/mL
50	**Insensitive Parameters**	BW				
51		D				

```
%Enclosure 4 - Randomized Virutal Patient Data
%Creates Random patient data for three patients

% %show original cobelli patient data
run Diab_Parameter_Vector
run Parameter_initialization; %picks off values and creates variables
sim Cobelli_correct %runs Cobelli Simulink model
 figure (1)
plot(time,gluc)
clear time gluc


%change each parameter by a random percentage

%set acceptable limits for the different parameters (use 10% of difference
%between healthy and diabetic in cobelli paper) %corrected to be centered
%around diabetic value
limit = [0.026174497
0.054761905
0.011267606
0.028571429
0.026022305
0.025
0.049868074
0.02808321
0.027881041
0.042205323
0.020288248
0
0.02
0.005263158
0.147826087
0.02
0
0.116666667
0.020588235
0.002608696
0.088888889
0.012621359
0.2
0.08
0.021374046
0.01969697
0
0.046236559
0.0265625
0.051611935
0.060595238
0.132323232
0.284615385
0.12
0
0
0
];



for p = 1:3
    change = zeros(37,1);
for i = 1:37

    change(i) = (2*(limit(i)).*rand(1,1)-limit(i)); %creates a vector of 35 random

end



New_Param = zeros(length(Parameters),1);


for i = 1:37
        New_Param(i) = (1+change(i))*Parameters(i); %modify all parameters, using
 end

Parameters = New_Param; %reset parameters vector to randomized parameters

run Parameter_initialization; %initialize parameters again with new values
sim Cobelli_correct

figure (p+1)
plot(time, gluc);
xlabel('Time (min)');
ylabel('Plasma Glucose Concentration (mg/dl)');

if p==1
    savefile = 'Patient1_3meals_correct.mat';
    title('Plasma Glucose: Patient 1 Parameters');
end
if p==2
    savefile = 'Patient2_3meals_correct.mat';
       title('Plasma Glucose: Patient 2 Parameters');
end
if p==3
    savefile = 'Patient3_3meals_correct.mat';
       title('Plasma Glucose: Patient 3 Parameters');
end

save(savefile,'Parameters', 'gluc', 'time')  %store glucose response



end

%plot results
```

```
clear all

hold on
load Patient1_3meals_Correct
plot(time, gluc,'r')
hold on
xlabel('Time (min)')
ylabel('Gluc (mg/dL)')
clear time gluc
load Patient2_3meals_Correct
plot(time,gluc,'g')
clear time gluc
load Patient3_3meals_Correct
plot(time,gluc, 'b')
legend('Patient 1', 'Patient 2', 'Patient 3');
```

```
        Error using Randomized_Data_Corrected (line 7)
        Unable to load block diagram 'Cobelli_correct'
```

Published with MATLAB® R2014a

```matlab
% Enclosure 5 Part 1- Parameter sensitivity testing code

% changes each parameter one a time, runs cobelli model,
% and calculates root mean squared error in the glucose output
% (a similar code computes and saves the error based on insulin data)

run Parameter_vector; % creates normal parameter vector and defines each specific
run Parameter_initialization; %picks off values and creates variables
sim Cobelli2 %runs Cobelli Simulink model
savefile = 'init_var.mat';
save(savefile, 'gluc', 'time')  %store original glucose response
init_gluc = gluc;


%change each parameter one at a time

changes = [0.1, 0.2, 0.3, 0.4 ,0.5, 0.6, 0.7, 0.8, 0.9, 1]; %change by 10% increme
results = zeros(length(ins), ((length(changes)*length(Parameters))+1)); %create sp
results(:,1) = time;
New_Param = zeros(length(Parameters),length(changes));
Sens = zeros((length(Parameters)-2),length(changes));
for i = 1:(length(Parameters)-2)
    for j = 1:length(changes)
        try
        New_Param(i,j) = (1+changes(j))*Parameters(i); %modify one parameter at a
        %creates a matrix where each row represents a different parameter
        %and the columns show the changes made
        run Parameter_vector; %reset parameter vector from previous changes
        Parameters(i) = New_Param(i,j); %reset parameter vector with new value
        run Parameter_initialization; %initialize parameters again
        sim Cobelli2 %run Cobelli model with new parameters
        results(:, ((i-1)*10+j+1)) = ins; %create matrix of glucose results: 1st c
        gluc_error = 100*sqrt(sum((gluc-init_gluc).^2)/length(gluc))/(max(gluc)-mi
        Sens(i,j) = gluc_error/changes(j); %similar to New_Param - shows sensitivi
        i
        catch
            disp('Oh No! An error occured - execution will continue') % use a 2 wh
            Sens(i,j) = 2;
        end
    end
end


savefile = 'Results.mat';
save(savefile, 'results', 'Sens')  %save glucose results in matlab file

        Error using run (line 55)
        Parameter_vector not found.

        Error in Encl_4_Parameter_Sensitivity (line 7)
        run Parameter_vector; % creates normal parameter vector and defines each s
```

Published with MATLAB® R2014a

```
%Enclosure 5 Part 2: Analyzing Sensitivity Results
%combines parameter sensitivity based on both insulin and glucose output data

%load results from parameter sensitivity code
load('C:\Users\m151656\Documents\MATLAB\Sens_ins_full.mat')
I = Sens;
load('C:\Users\m151656\Documents\MATLAB\Sens_gluc_full.mat')
G = Sens;

% find the average sensitivity for each insulin and glucose
maximum_I = zeros(35, 2);
avg_I = zeros(35,2);
for i = 1:35
    maximum_I(i,1) = i;
    avg_I(i,1) = i;
    maximum_I(i,2) = max(I(i,:));
    avg_I(i,2) = mean(I(i,:));
end

maximum_G = zeros(35, 2);
avg_G = zeros(35,2);
for i = 1:35
    maximum_G(i,1) = i;
    avg_G(i,1) = i;
    maximum_G(i,2) = max(G(i,:));
    avg_G(i,2) = mean(G(i,:));
end

%sort results
maximum_I(:, 3:4) = sortrows(maximum_I,-2); %columns 3 and 4 will show sorted valu
avg_I(:, 3:4) = sortrows(avg_I,-2); %columns 3 and 4 will show sorted values
maximum_G(:, 3:4) = sortrows(maximum_G,-2); %columns 3 and 4 will show sorted valu
avg_G(:, 3:4) = sortrows(avg_G,-2); %columns 3 and 4 will show sorted values

%combine insulin and glucose sensitivity
Comb_max = zeros(35,2);
Comb_avg = zeros(35,2);
for i = 1:35
    Comb_max(i,1) = i;
    Comb_avg(i,1) = i;
    Comb_max(i,2) = sqrt((maximum_G(i,2))^2+(maximum_I(i,2))^2);
    Comb_avg(i,2) = sqrt((avg_G(i,2))^2+(avg_I(i,2))^2);
end

%sort results
Comb_max(:, 3:4) = sortrows(Comb_max,-2); %columns 3 and 4 will show sorted values
Comb_avg(:, 3:4) = sortrows(Comb_avg,-2); %columns 3 and 4 will show sorted values

%save in excel and label columns
savefile = 'Sensitivity_Results_Gluc_Ins';
col_header = {'Parameter Number', 'Max Sensitivity','Parameter Number', 'Max Sensi
row_header(1:35,1)  = {'Vg', 'k1', 'k2', 'ke1', 'ke2', 'VI', 'm1', 'm2', 'm4', 'm5
```

```
xlswrite(savefile, Comb_max, 1, 'B2');
xlswrite(savefile, col_header, 1, 'B1');
xlswrite(savefile, row_header, 1, 'A2');

col_header = {'Parameter Number', 'Avg Sensitivity','Parameter Number', 'Avg Sensi
xlswrite(savefile, Comb_avg, 2, 'B2');
xlswrite(savefile, col_header, 2, 'B1');
xlswrite(savefile, row_header, 2, 'A2');
```

```
        Error using load
        Unable to read file 'C:\Users\m151656\Documents\MATLAB\Sens_ins_full.mat':

        Error in Encl_4_Sensitivity_Analysis (line 5)
        load('C:\Users\m151656\Documents\MATLAB\Sens_ins_full.mat')
```

Published with MATLAB® R2014a

```
%Enclosure 6 Part 1: Parameter Adaptation Main File

% This is the top-level script.  It calls the optimization
function fmincon and makes all variable declarations necessary
for this call.  Those variable declarations include: 1) The
function handle for the "management" function (the function to
be called), 2) The upper and lower bounds for each parameter
being fit, 3) Initial condition vector for the parameters.
These elements must be placed in the appropriate "spot" in
% the fmincon function call.  (Type "help fmincon" in Command
Window to see full documentation and order of arguments.  Square
braces [] are used to give no input to a specific function
argument.  Order matters, and  fmincon assigns inputs to each
argument in the order received, so use [] to skip specific
arguments and enter later ones.)

% Function handle for function to be called
f = @(p)Dilks_model_sim_10(p);

% Lower and upper bounds of parameters (10 parameters to be
fitted:  Vg,
%  k1, VI,m4, m5, m6, kabs, kgri, kp1, kp4) (1, 2, 6, 9, 10, 11,
15, 16, 22, 25)
LB = [1.451 0.0397 0.039 0.2615 0.05038 0.79533 0.0196 0.04557
3.051 0.07692];
UB = [1.529 0.0443 0.041 0.2765 0.05482 0.82827 0.0264 0.04743
3.129 0.08028];

% Best initial guess
p0 = [1.49 0.042 0.04 0.269 0.0526 0.8118 0.023 0.0465 3.09
0.0786];

% Calling the optimization function
[P cost] = fmincon(f,p0,[],[],[],[],LB,UB)

save Parameter_Adaptation_Results_new P cost
```

```
% Enclosure 6 Part 2: Management Function (adapts parameters)

% This is the "management" function.  It handles passing the
candidate
% parameter vector to the model and computing the cost based on
the
% returned model simulation.
%

function F = Dilks_model_sim_10(p)

run Diab_Parameter_Vector %load original Diabetic Cobelli
parameter values
index = [1, 2, 6, 9, 10, 11, 15, 16, 22, 25]; %specific
parameters to be updated
for i = 1:length(index)
Parameters(index(i)) = p(i); %update only the parameters we're
interested in
end
run Param_Adap_Setup %assigns new parameter values and creates
new Initial conditions
%uses patient data to create time vector for data points
Parameters(38) = Ib;
Parameters(39) = Sb;
Parameters(40) = h;
Parameters(41) = meal;
% Printing the parameter vector chosen by fmincon each
interation
p

% Solving the differential equation (running the model)
[time, Cstates] =
ode45(@Cobelli,T_data,InitCond',[],Parameters);
gluc = Cstates(:,9)/Vg;


% Calculate the cost of this parameter vector, p, by comparing
simulation
% to data at the same time points.  Print this value each
iteration.
F = sum((gluc-data).^2);
```

```
%Enclosure 6 Part 3: Setup file for Parameter Adaptation
%Sets up parameter variables, initial conditions, and data time
vector

%Pick off Parameters for use in Simulink Model
Tstep = 0.01;
%   Glucose Kinetics
Vg = Parameters(1);
k1 = Parameters(2);
k2 = Parameters(3);
%   Renal Excretion
ke1 = Parameters(4);
ke2 = Parameters(5);
%   Insulin Kinetics
Vi = Parameters(6); %note: Vi in Weisberg
m1 = Parameters(7);
m2 = Parameters(8);
m4 = Parameters(9);
m5 = Parameters(10);
m6 = Parameters(11);
HEb = Parameters(12); % where is this used?
%   Rate of Appearance - Gastro Intestinal Tract
kmax = Parameters(13);
kmin = Parameters(14);
kabs = Parameters(15);
kgri = Parameters(16);
f = Parameters(17);
a = Parameters(18);
b = Parameters(19);
c = Parameters(20);
d = Parameters(21);
%   Endogenous Production - Liver
kp1 = Parameters(22);
kp2 = Parameters(23);
kp3 = Parameters(24);
kp4 = Parameters(25);
ki = Parameters(26);
%   Utilization - muscle and adipose tissue
Fcns = Parameters(27);
Vm0 = Parameters(28);
Vmx = Parameters(29);
Km0 = Parameters(30);
p2U = Parameters(31);
%   Secretion
K = Parameters(32);
alpha = Parameters(33);
Beta = Parameters(34); %note: different than weisberg
```

```
gamma = Parameters(35); %note: different than weisberg
%h = Parameters(36);
%   Insensitive Parameters
%BW = Parameters(37);
%D = Parameters(38);

% Prof O'Brien change 18 Nov
BW = Parameters(36);
D = Parameters(37);


%Basal Parameters for Cobelli Model: (initial conditions)
    %  Insulin
Sb = (m6 - HEb)/m5;
m3b = (HEb*m1)/(1-HEb);
matrix = [m1+m3b, -m2; -m1, m2+m4];
i_matrix = matrix^-1*[Sb; 0];
Ilb = i_matrix(1);
Ipb = i_matrix(2);
Ib = Ipb/Vi;
Ipob = Sb/gamma;

 %  Glucose
 Ipob = Sb/gamma;
 syms Gtb Gpb EGPb %recognizes unknown variables as symbols
 %assuming Gpb > Ke2
 S1 = 'kp1 == EGPb + kp2*Gpb+kp3*Ib+kp4*Ipob'; %Equation 12
 S2 = 'EGPb == Fcns + (Vm0*Gtb)/(Km0+Gtb) + ke1*(Gpb - ke2)';
%Equation 14, 15, 27
 S3 = '(Vm0*Gtb)/(Km0+Gtb) = k1*Gpb - k2*Gtb'; %Equation 1
 S_Gluc = solve(S1, S2, S3, Gtb, Gpb, EGPb); %returns a
structure with solved equations
 tmp = double(subs(S_Gluc.Gtb)); %substitute in known parameters
 Gtb = tmp(2); % pick out positive solution
 tmp =  double(subs(S_Gluc.Gpb));
 Gpb = tmp(2);
 tmp = double(subs(S_Gluc.EGPb));
 EGPb = tmp(2);

 if (Gpb<=ke2) %checks if Gpb < ke2 - if so, need to change
equations
     clear Gtb Gpb EGPb;
     syms Gtb Gpb EGPb;
     S1 = 'kp1 == EGPb + kp2*Gpb+kp3*Ib+kp4*Ipob'; %Equation 12
     S2 = 'EGPb == Fcns + (Vm0*Gtb)/(Km0+Gtb)'; %Equation 14,
15, 27
     S3 = '(Vm0*Gtb)/(Km0+Gtb) = k1*Gpb - k2*Gtb'; %Equation 1
     S_Gluc = solve(S1, S2, S3, Gtb, Gpb, EGPb); %returns a
structure with solved equations
     tmp = double(subs(S_Gluc.Gtb)); %substitute in known
parameters
     Gtb = tmp(2); % pick out positive solution
     tmp =  double(subs(S_Gluc.Gpb));
     Gpb = tmp(2);
     tmp = double(subs(S_Gluc.EGPb));
     EGPb = tmp(2);
 end

 % Prof O'Brien edit 18 Nov
 Gb = Gpb/Vg;
 h = Gb;

 % meal at t=1
 meal = 1;

InitCond = zeros(12,1); %initial values for all states
InitCond(1) = D; %Qsto1
InitCond(2) = 0; %Qsto2
InitCond(3) = 0; %Qgut
InitCond(4) = Ib; %I1
InitCond(5) = Ib; %Id
InitCond(6) = Ipb; %Ip
InitCond(7) = Ipob; %Ipo - find this!
InitCond(8) = 0;%X
InitCond(9) = Gpb; %Gp
InitCond(10) = Gtb; %Gt
InitCond(11) = 0; %Y
InitCond(12) = Ilb; %Il

load Patient1_correct_data % load random patient data
%record what the original patient parameters were for
verification
%original parameters: 1.4636, 0.0408, 0.0394, 3.0783, 0.0798
% pick off specific time points from the data - glucose readings
every 20
% min (15 readings total)
T_data = zeros(16,1);
data = zeros(16,1);
T_data(1) = pat_time(1);
data(1) = pat_gluc(1);
for i = 1:15
    T_data(i+1) = pat_time(i*20*1000);
    data(i+1) = pat_gluc(i*20*1000);
end
```

```
%Enclosure 6 Part 4: Cobelli Model Equations
%runs the cobelli model using an ODE solver


%inputs:
%t=current time
%state=current state of tracked variables
%Parameters=parameter vector
%inputs = 2 row vector of insulin and glucagon injection amounts



function stateReturn = Cobelli(t, state, Parameters)


%% plucks off parameters, assigns variable names, creates
initial conditions
%Pick off Parameters for use in Cobelli model

%   Glucose Kinetics
Vg = Parameters(1);
k1 = Parameters(2);
k2 = Parameters(3);
%   Renal Excretion
ke1 = Parameters(4);
ke2 = Parameters(5);
%   Insulin Kinetics
Vi = Parameters(6);
m1 = Parameters(7);
m2 = Parameters(8);
m4 = Parameters(9);
m5 = Parameters(10);
m6 = Parameters(11);
HEb = Parameters(12);
%   Rate of Appearance - Gastro Intestinal Tract
kmax = Parameters(13);
kmin = Parameters(14);
kabs = Parameters(15);
kgri = Parameters(16);
f = Parameters(17);
a = Parameters(18);
b = Parameters(19);
c = Parameters(20);
d = Parameters(21);
%   Endogenous Production - Liver
kp1 = Parameters(22);
kp2 = Parameters(23);
```

```
kp3 = Parameters(24);
kp4 = Parameters(25);
ki = Parameters(26);
%   Utilization - muscle and adipose tissue
Fcns = Parameters(27);
Vm0 = Parameters(28);
Vmx = Parameters(29);
Km0 = Parameters(30);
p2u = Parameters(31);
%   Secretion
K = Parameters(32);
alpha = Parameters(33);
Beta = Parameters(34);
gamma = Parameters(35);

% Prof O'Brien change 18 Nov
BW = Parameters(36);
D = Parameters(37);
Ib = Parameters(38);
Sb = Parameters(39);
h = Parameters(40);
meal = Parameters(41);

 %% State Vector
%Rate of Appearance
Qsto1 = state(1); %where meal input is implemented as initial
condition
Qsto2 = state(2);
Qgut = state(3);
I1 = state(4);
Id = state(5); %delayed insulin signal
Ip = state(6);%plasma insulin concentration
Ipo = state(7);
X = state(8); %interstitial insulin (muscle and adipose)
Gp = state(9);
Gt = state(10);
Y = state(11);
Il = state(12);


 %% Model Equations %does the order of these equations matter?
  %Rate of Appearance
Qsto=Qsto1+Qsto2;
KemptQsto=kmin+(kmax-kmin)/2*(tanh(a*(Qsto-b*D))-tanh(c*(Qsto-
d*D))+2);
dQsto1=-kgri*Qsto1;
dQsto2=-KemptQsto*Qsto2+kgri*Qsto1;
```

```
dQgut=-kabs*Qgut+KemptQsto*Qsto2;
Ra=f*kabs*Qgut/BW;

%Liver
I=Ip/Vi;
dI1=-ki*(I1-I);
dId=-ki*(Id-I1);
EGP=kp1-kp2*Gp-kp3*Id-kp4*Ipo;

%Muscle and Adipose Tissue
dX=-p2u*X+p2u*(I-Ib);
VmX=Vm0+Vmx*X;
Uid=VmX*Gt/(Km0+Gt);
Uii=Fcns;

%Glucose Kinetics
if (Gp>ke2)
    E=ke1*(Gp-ke2);
else
    E=0;
end
dGp=EGP+Ra-Uii-E-k1*Gp+k2*Gt;
dGt=-Uid+k1*Gp-k2*Gt;
G=Gp/Vg;
dG=dGp/Vg;

%Pancreas
    %static secretion
if(Beta*(G-h)>=-Sb)
    dY=-alpha*(Y-Beta*(G-h));
else
    dY=-alpha*Y-a*Sb;
end

    %dynamic secretion
if(dG>0)
    Spo=Y+K*dG+Sb;
else
    Spo=Y+Sb;
end

dIpo=-gamma*Ipo+Spo;
S=gamma*Ipo;

%Insulin Dynapics
HE=-m5*S+m6;
m3=HE*m1/(1-HE);
```

```
dIl=-(m1+m3)*Il+m2*Ip+S;
dIp=-(m2+m4)*Ip+m1*Il;


%% return
stateReturn=[dQsto1;dQsto2;dQgut;dI1;dId;dIp;dIpo;dX;dGp;dGt;dY;
dIl];
end
   %this function returns state derivatives and uses them and
the current
   %state to determine the next state
```